

DOCKET NO: 250727US8

IN THE UNITED STATES PATENT & TRADEMARK OFFICE

IN RE APPLICATION OF :
JOHN MARTIN, ET AL. : EXAMINER: PHAM, M.
SERIAL NO: 10/821,949 :
FILED: APRIL 12, 2004 : GROUP ART UNIT: 2167
FOR: ELECTRONIC DISCOVERY :
APPARATUS, SYSTEM, METHOD, AND
ELECTRONICALLY STORED
COMPUTER PROGRAM PRODUCT

PROPOSED EXAMINER'S AMENDMENT

IN THE SPECIFICATION

[0077] The term "computer readable medium" as used herein refers to any medium that participates in providing instructions to the processor 1203 for execution. A computer readable storage medium may take many forms, including but not limited to, non-volatile media, and volatile media. Non-volatile media includes, for example, optical, magnetic disks, and magneto-optical disks, such as the hard disk 1207 or the removable media drive 1208. Volatile media includes dynamic memory, such as the main memory 1204. Transmission media includes coaxial cables, copper wire and fiber optics, including the wires that make up the bus 1202. Wireless transmission media also may also take the form of acoustic or light waves, such as those generated during radio wave and infrared data communications.

IN THE CLAIMS

Claims 1-2. (Canceled).

Claim 3. (Previously Presented) The method of Claim 49, wherein said converting step includes identifying a file that is not convertible as an exception file.

Claim 4. (Previously Presented) The method of Claim 49, further comprising a step of:

displaying summary statistics prior to said outputting step.

Claim 5. (Previously Presented) The method of Claim 4, wherein said step of displaying comprises:

displaying a series of parent-child relationships between a file and an attachment to said file said parent-child relationship being a document context.

Claim 6. (Previously Presented) The method of Claim 5, wherein at least said converting step is a step performed in a software plug-in module.

Claim 7. (Previously Presented) The method of Claim 49, wherein said metadata comprises:

date created, last date opened, last date modified, creator name, matter name, and predetermined identification and quality control data.

Claim 8. (Cancelled).

Claim 9. (Previously Presented) The method of Claim 49, wherein said step of converting further comprises: marking a file that cannot be converted as an exception file.

Claim 10. (Previously Presented) The method of Claim 49, wherein said step of physically loading and copying comprises:

transferring via a drag-and-drop user interface comprising one of a computer mouse and a pointing device.

Claim 11. (Previously Presented) The method of Claim 49, wherein said working copy comprises:

a time-stamped audit file configured to record a file history spanning file creation to file destruction.

Claim 12. (Cancelled)

Claim 13. (Currently Amended) The method of Claim 49 ^{[[12]]}, further comprising: performing page counts and time stamping/digital authentication of said output file.

Claim 14. (Cancelled).

Claim 15. (Previously Presented) The method of Claim 9, wherein said file that cannot be converted comprises one of:

a file with a virus;

an encrypted file;

a corrupted file;

an unknown file-type; and
a deselected file.

Claim 16. (Previously Presented) The method of Claim 9, wherein said step of marking a file that cannot be converted comprises one of:

logging said exception file; and
exporting said exception file.

Claim 17. (Cancelled)

Claim 18. (Cancelled)

Claim 19. (Previously Presented) The method of Claim 49, wherein said step of converting comprises:

time stamping and digitally authenticating both an image and a file of extracted meta-data.

Claim 20. (Previously Presented) The method of Claim 49, wherein said step of converting comprises one of:

time stamping or digitally authenticating the output file; and
appending selected meta-data.

Claim 21. (Previously Presented) The method of Claim 20, wherein said step of converting comprises:

creating one or more searchable subordinate text files containing the contents of an operator-selected subset of the selected files;

time stamping or digitally authenticating the one or more subordinate text files; and
appending selected meta-data about the files included in the subordinate text files.

Claim 22. (Previously Presented) The method of Claim 49, further comprising:
extracting one of file content data, content header information, file meta-data, file type information, and file characteristic data.

Claim 23. (Previously Presented) The method of Claim 49, wherein said associated meta-data comprises:

predetermined categories of meta-data corresponding to a file-type.

Claim 24. (Previously Presented) The method of Claim 19, wherein said step of time stamping comprises:

time stamping with one of UTC time and another predetermined time zone.

Claim 25. (Previously Presented) The method of Claim 49, wherein said step of extracting text comprises:

creating an ASCII file of said subset of text.

Claim 26. (Previously Presented) The method of Claim 25, wherein said step of extracting text further comprises:

converting said image of said selected file to text with an OCR program if an ASCII file cannot be created.

Claim 27. (Previously Presented) The method of Claim 49, wherein said step of extracting text comprises:

searching for a key word.

Claim 28. (Previously Presented) The method of Claim 49, wherein said step of extracting text comprises:

extracting a portion of text around said keyword.

Claim 29. (Previously Presented) The method of Claim 49, wherein said step of processing and converting comprises:

processing with a prioritization scheme keyed to file type.

Claim 30. (Previously Presented) The method of Claim 29, wherein said step of processing with a prioritization scheme comprises:

processing by one of file extension and file header with a plug-in module.

Claim 31. (Previously Presented) The method of Claim 30, wherein said step of processing with a plug-in module comprises:

processing with a plug-in module configured to be selected to be 'ON' or 'OFF.'

Claim 32. (Previously Presented) The method of Claim 30, wherein said step of processing with a plug-in module comprises;

processing with a plug-in module configured to have a user-selectable priority.

Claim 33. (Previously Presented) The method of Claim 29, wherein said step of processing with a prioritization scheme keyed to file type comprises at least one of:

- identifying a file type extension; and
- evaluating a binary file header.

Claim 34. (Previously Presented) The method of Claim 33, wherein said step of processing with a prioritization scheme keyed to file type comprises one of:

- identifying said file type extension and evaluating the binary header when the file type extension is unknown; and

- evaluating the binary header, and if there is a conflict between the binary header and the file-type extension, one of the binary header or the file-type extension is considered a default first choice, either arbitrarily or based on a predetermined logic keyed to a predetermined file type.

Claim 35. (Previously Presented) The method of Claim 30, wherein said step of processing with a plug-in module comprises one of:

- processing with a plug-in configured to open multiple file types;
- processing with multiple plug-ins configured to open a single file-type;
- processing with a plug-in created by incorporating a library of a commercially available software product; and
- processing with a plug-in comprising a library of programming code that incorporates functionality of a third party library or an application to load, image and extract metadata from a document.

Claim 36. (Currently Amended) The method of Claim 29, wherein said step of processing with a prioritization scheme keyed to file type comprises:

reading a file that is correlated to a plug-in.

Claim 37. (Previously Presented) The method of Claim 36, wherein said step of processing with a prioritization scheme keyed to a file type comprises:

reading a file that cannot be processed by the plug-in by using a Microsoft Windows File Type Association and accessing a Windows registry to determine if a "print" verb is associated with the extension in Windows;

if said "print" verb is found to be associated with the extension, starting a new Windows process with said "print" verb as startup information and feeding the output of said new Windows process to an imaging print driver.

Claim 38. (Previously Presented) The method of Claim 49, wherein said step of converting comprises:

imaging with an imaging driver.

Claim 39. (Previously Presented) The method of Claim 3, wherein said step of outputting comprises:

exporting to at least one of an image viewer, a printer, a computer and another media.

Claim 40. (Previously Presented) The method of Claim 39, wherein said step of outputting comprises:

exporting the output file comprising a predetermined litigation support software file type.

Claim 41. (Previously Presented) The method of Claim 40, wherein said predetermined litigation support software file type comprises one of:

an IPRO file type, an Opticon file type, a Concordance file type, a Summation file type, a Ringtail file type, a Microsoft Access file type, and other data management file types.

Claims 42-43. (Canceled).

Claim 44. (Previously Presented) The method of Claim 49, further comprising: pre-filtering and saving pre-filtering criteria.

Claim 45. (Previously Presented) The method of Claim 49, wherein said step of physically loading and copying comprises:

accessing data from an email or instant messaging archive.

Claim 46. (Previously Presented) The method of Claim 45, wherein said step of accessing data from an email or instant message archive comprises:

accessing a printable attachment of an email or instant message.

Claims 47-48. (Canceled).

Claim 49. (Currently Amended) A computer implemented method of processing and converting electronically-stored data for electronic discovery in support of litigation, comprising:

physically loading and copying data and associated meta-data into a processor-based device by a user at a user-site from one or more data storage devices located at [[a]] the user-site, said processor-based device being located at said user-site;

inputting from the user on-site user input to said processor-based device for subsequent processing at a user-site of a working copy of the data and associated meta-data, said subsequent processing including storing in a local storage device said data and associated meta-data as the working copy for file processing, while maintaining a document context with respect to other documents, said working copy of the data including a plurality of files;

said file processing of the plurality of files in the working copy of the data including identifying a duplication of one or more files within the working copy of the data, including

- identifying a duplication by a hash algorithm,
- checking a duplicate file for file corruption, and
- exporting a duplicated file, and

converting a selected file to an output file in a user-specified format with said processor-based device if said selected file is convertible to the user-specified format, said step of converting including

- extracting and saving file meta-data associated with said selected file,
- extracting text from said selected file, and

creating an image of said selected file, at least one of

- checking for and removing a virus from one or more files, and
- checking for encoding or encryption of one or more files, and decoding or decrypting said one or more files if encoded or encrypted; and

outputting the output file in the user-specified export format for at least one of exporting, reviewing and searching the output file in an external system, wherein said steps of physically loading and copying, inputting, converting, processing and converting, and outputting being performed by said processor-based device at said user-site.

Claim 50. (Currently Amended) A system for processing and converting electronically-stored data for electronic discovery in support of litigation, comprising:

a data input device configured to allow a user to physically load and copy data and associated meta-data at a user-site from one or more data storage devices located at ~~[[a]]~~ the user-site into a processor-based device located at said user-site;

an interface configured to receive on-site user input into said processor-based device for ~~subsequently~~ subsequent file processing at the user-site of a working copy of the data and associated meta-data;

a processor configured to store in a local storage device said data and associated meta-data as the working copy, while maintaining a document context with respect to the other documents, said working copy of the data including a plurality of files,

said file processing of the plurality of files in the working copy of the data including
identifying a duplication of one or more files within the working copy of the data, including

identifying a duplication by a hash algorithm,
checking a duplicate file for file corruption, and
exporting a duplicated file, and

said processor also configured to convert a selected file to an output file in a user-specified format if said selected file is convertible to a user-specified format, said converting including

extracting and saving file meta-data associated with said selected file,

extracting text from said selected file, and
creating an image of said selected file, said processor being inside said
processor-based device, and
at least one of

checking for and removing a virus from one or more files, and
checking for encoding or encryption of one or more files, and decoding or
decrypting said one or more files if encoded or encrypted; and

an output mechanism configured to output the output file in the user-specified export
format for at least one of exporting, reviewing and searching the output file in an external
system, wherein said data input device, said interface, and said processor inside said
processor-based device being located on said user-site.

Claim 51. (Currently Amended) A computer ~~program product~~ readable storage
medium including instructions ~~configured to that when executed by a computer control~~ [[a]]
the computer so as to process and convert electronically-stored data for electronic discovery
in support of litigation, when instructions of said computer program product are executed by
said computer, said computer is caused to perform a process comprising steps of:

physically loading and copying data and associated meta-data into a processor-based
device by a user at a user-site from one or more data storage devices located at [[a]] the user-
site, said processor-based device being located at said user-site;

inputting from the user on-site user input to said processor-based device for
subsequent processing at the user-site of a working copy of the data and associated meta-data,
said subsequent processing including

storing in a local storage device said data and associated meta-data as the working copy for file processing, while maintaining a document context with respect to other documents, said working copy of the data including a plurality of files;

said file processing of the plurality of files in the working copy of the data including identifying a duplication of one or more files within the working copy of the data, including

identifying a duplication by a hash algorithm,
checking a duplicate file for file corruption, and
exporting a duplicated file, and

converting a selected file to an output file in a user-specified format with said computer if said selected file is convertible to the user-specified format, said step of converting including

extracting and saving file meta-data associated with said selected file,
extracting text from said selected file, and
creating an image of said selected file,

at least one of

checking for and removing a virus from one or more files, and
checking for encoding or encryption of one or more files, and decoding or decrypting said one or more files if encoded or encrypted; and

outputting the output file in the user-specified export format for at least one of exporting, reviewing and searching the output file in an external system, wherein said steps of physically loading and copying, inputting, converting, processing and converting, and outputting being performed by said processor-based device at said user-site.